UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/875,289	06/07/2001	Steven Roth	10004466-2	4656

7590 03/22/2005

HEWLETT-PACKARD COMPANY
IP Administration
Legal Department, M/S 35
P.O. Box 272400
Fort Collins, CO 80528-9599

EXAMINER	
CHANG, ERIC	

ART UNIT	PAPER NUMBER
2116	

DATE MAILED: 03/22/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/875,289	ROTH ET AL.	
	Examiner	Art Unit	
	Eric Chang	2116	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *27 December 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-30 are pending.

Claim Rejections - 35 USC § 102

2. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

3. Claims 1-30 are rejected under 35 U.S.C. 102(b) as being clearly anticipated by the preface of "AIX for Breakfast" by Houtz.

4. As to claim 1, Houtz discloses a method of updating tunables used in a kernel, comprising: updating a system file including tunables each having a tunable setting with a new tunable value in response to a single administrator request [paragraphs 2-4]; simultaneously updating a persistent storage mechanism including tunables each having a tunable setting with the new tunable value in response to the single administrator request [paragraph 10]; changing a tunable value in the kernel with the new tunable value and continuing to run the computer with the updated tunable value [paragraph 6]. Houtz teaches that kernel parameters, such as tunables or other configuration information, are traditionally changed manually by editing configuration data files [paragraph 4]. Houtz further teaches using a System Management Interface Tool (SMIT) to change each parameter by a single request in a series of reconfiguration steps [paragraph 2], and that the kernel continues to run using the updated value [paragraph 6]. In addition, Houtz teaches that SMIT performs the actual editing of files and linkages [paragraph

3]; therefore, the new value is updated to a persistent storage mechanism, substantially as claimed. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

5. As to claim 2, Houtz discloses updating steps are performed using SMIT, a system administrator's management application [paragraph 2].

6. As to claim 3, Houtz discloses updating steps are performed using a UNIX command line [paragraph 6]. Houtz teaches reconfiguring by issuing a command with parameters.

7. As to claim 4, Houtz discloses the computer includes a UNIX operating system [paragraph 2].

8. As to claim 5, Houtz discloses the system file includes a file for a core kernel and another for each separately loadable kernel module [paragraph 6]. Houtz teaches files for a core kernel and for each service in the operating system.

9. As to claim 6, Houtz discloses compiling the kernel and incorporating the new tunable value into the kernel [paragraph 6].

10. As to claim 7, Houtz discloses the persistent storage mechanism is a Kernel Registry Service [paragraph 6]. Houtz teaches a database for storing key elements and devices, substantially as claimed.

11. As to claims 8-9, Houtz discloses editing system files to change existing parameters [paragraphs 2-4]; it would be well known to one of ordinary skill in the art that this comprises retrieving and saving data from said existing system files. In addition, Houtz teaches that the changes are made by using a menu system [paragraph 2], and that descripting information about kernel and operating system parameters are stored in a database [paragraph 6]; thus, Houtz teaches retrieving enough detailed information about one or more tunable settings in order to provide a meaningful user interface for changing said settings.

12. As to claim 10, Houtz discloses registering handler functions for a tunable handler for a particular tunable [paragraph 2]. Houtz teaches using a menu task in SMIT to edit a particular tunable.

13. As to claim 11, Houtz discloses interfacing with the kernel using a handler function [paragraphs 6 and 10]. Houtz teaches using the SMIT interface to change the kernel.

14. As to claims 12-13, Houtz discloses a method of rebuilding a kernel, comprising: retrieving tunable settings stored in a system file [paragraphs 2-4]; rebuilding the kernel using the retrieved tunable settings [paragraph 6]; and updating a persistent storage mechanism using

the retrieved tunable settings [paragraph 3]. Houtz teaches editing system files, to change existing parameters; it would be well known to one of ordinary skill in the art that this comprises retrieving, saving and/or overwriting data from said existing system files. Furthermore, Houtz teaches rebuilding and using the updated parameters, substantially as claimed. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

15. As to claim 14, Houtz discloses a dynamic kernel tunable framework for changing tunables in a kernel without rebooting, comprising: a graphical user interface for displaying and changing graphical values and settings of dynamic tunables [paragraphs 2 and 6]; a system call interface for interfacing the user interface with a system file, a persistent storage mechanism and the kernel [paragraph 6]; a handler function interface interfaced to the system call interface and the kernel including information about each dynamic tunable [paragraph 6]. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

16. As to claim 15, Houtz discloses tunable changes are made immediately without rebooting and are also kept persistent across reboots, both actions taken as the result of a single

administrator request [paragraph 6]. Houtz teaches the changes are made immediately without rebooting the kernel, and that the files are likewise updated with the new information [paragraph 2]. Furthermore, the actions are taken as a result of accepting the new values for the parameters [paragraph 2], substantially as claimed.

17. As to claim 16, Houtz discloses a computer architecture, comprising: updating means for updating a system file including tunables each having tunable setting with a new tunable value in response to a single administrator request [paragraph 2]; updating means for simultaneously updating a persistent storage mechanism including tunables each having tunable setting with the new tunable value in response to the single administrator request [paragraph 3]; and changing means for changing a tunable value in the kernel with the new tunable value and continuing to run the computer with the updated tunable value [paragraph 6]. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

18. As to claim 17, Houtz discloses computer architecture, comprising: retrieving tunable settings stored in a system file [paragraphs 2-4]; rebuilding the kernel using the retrieved tunable settings [paragraph 6]; and updating a persistent storage mechanism using the retrieved tunable settings [paragraph 3]. Houtz teaches editing system files, to change existing parameters; it would be well known to one of ordinary skill in the art that this comprises retrieving, saving and/or overwriting data from said existing system files. Furthermore, Houtz teaches rebuilding

and using the updated parameters, substantially as claimed. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

19. As to claim 18, Houtz discloses an article, comprising: a medium bearing the executable instructions in machine readable form, comprising instructions to: retrieve a system file including tunables each having tunable setting with a new tunable value in response to a single administrator request [paragraphs 2-3]; simultaneously retrieve a persistent storage mechanism including tunables each having tunable setting with the new tunable value in response to the single administrator request [paragraphs 2-3]; and change a tunable value in the kernel with the new tunable value and continuing to run the computer with the retrieved tunable value [paragraph 6]. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

20. As to claim 19, Houtz discloses an article, comprising: a medium bearing the executable instructions in machine readable form comprising instructions to: retrieving tunable settings stored in a system file [paragraphs 2-4]; rebuilding the kernel using the retrieved tunable settings [paragraph 6]; and updating a persistent storage mechanism using the retrieved tunable settings [paragraph 3]. Houtz teaches editing system files, to change existing parameters; it would be

well known to one of ordinary skill in the art that this comprises retrieving, saving and/or overwriting data from said existing system files. Furthermore, Houtz teaches rebuilding and using the updated parameters, substantially as claimed. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

21. As to claim 20, Houtz discloses a computer system, comprising: a processor and a memory coupled to said processor, the memory having stored therein sequences of instructions, which, when executed by said processor, causes said processor to perform the steps of: update a system file including tunables each having tunable setting with a new tunable value in response to a single administrator request [paragraphs 2-3]; simultaneously update a persistent storage mechanism including tunables each having tunable setting with the new tunable value in response to the single administrator request [paragraphs 2-3]; and change a tunable value in the kernel with the new tunable value and continuing to run the computer with the updated tunable value [paragraph 6]. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

22. As to claim 21, Houtz discloses a computer system, comprising: a processor and a memory coupled to said processor, the memory having stored therein sequences of instructions,

which, when executed by said processor, causes said processor to perform the steps of: update tunable settings stored in a system file [paragraphs 2-4]; rebuild the kernel using the retrieved tunable settings [paragraph 6]; and update a persistent storage mechanism using the retrieved tunable settings [paragraph 3]. In addition, Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

23. As to claims 22-23, Houtz discloses that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically independently of each other and without needing any centralized interface changes, because only the code using the tunable needs to be changed.

24. As to claims 24-25, Houtz discloses a data structure including information about every tunable parameter for use in a registry database [paragraphs 17-18].

25. As to claims 26-30, Houtz discloses a using an associated handler function to update a tunable [paragraph 10]. It is well known to one of ordinary skill in the art that such a handler function comprises one of a SAM, kmtune or kernel system call, substantially as claimed.

Response to Arguments

26. Applicant's arguments filed December 27, 2004 have been fully considered but they are not persuasive.

27. In the remarks, applicants argued in substance that Houtz does not teach or suggest that the tunable value in the kernel is changed without needing any centralized interface changes. But Houtz teaches that the tunable value in the kernel is changed on a per-service basis without recompiling the entire kernel [paragraph 6]. Thus, each tunable can be made changed dynamically without needing any centralized interface changes, because only the code using the tunable needs to be changed.

Conclusion

28. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

29. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Chang whose telephone number is (571) 272-3671. The examiner can normally be reached on M-F 9:00-5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lynne Browne can be reached on (571) 272-3670. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

LYNNE H. BROWNE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

March 16, 2005
ec